UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/813,433	03/31/2004	Simon Knowles	66365-021	3801

7590 01/15/2009

MCDERMOTT, WILL & EMERY
600 13th Street, N.W.
Washington, DC 20005-3096

EXAMINER	
HUISMAN, DAVID J	
ART UNIT	**PAPER NUMBER**
2183	
MAIL DATE	**DELIVERY MODE**
01/15/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/813,433	Applicant(s) KNOWLES, SIMON	
	Examiner DAVID J. HUISMAN	Art Unit 2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *20 October 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-23* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-23* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *31 March 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *8/6/2008*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-23 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: RCE and IDS as received on 8/6/2008 and Amendment as received on 10/20/2008.

Specification

3. The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the following is required: The specification does not provide proper antecedent basis for "computer readable medium" as claimed in claim 23.

Information Disclosure Statement

4. The Hull reference (U.S. Patent No. 5,922,065) cited by applicant in the IDS filed on August 6, 2008, has not been considered because this exact reference was cited by the examiner in the 892 mailed on August 20, 2007.

Claim Objections

5. Claim 23 is objected to because of the following informalities: In the 8^{th} to last line, insert --a-- after "supply". Appropriate correction is required.

Claim Rejections - 35 USC § 112

6. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

7. Claims 22-23 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

8. Claim 22 recites the limitation "said data processing instruction" in the 3rd to last paragraph. There is insufficient antecedent basis for this limitation in the claim because applicant claims "at least one data processing instruction" Therefore, when there is more than one instruction, it is not clear which one instruction is "said data processing instruction".

9. Claim 22 recites the limitation "said execution path" in the last line. There is insufficient antecedent basis for this limitation in the claim because it is not clear which path applicant is referring to.

10. Claim 23 recites the limitation "said execution path" in the last line. There is insufficient antecedent basis for this limitation in the claim because it is not clear which path applicant is referring to.

Claim Rejections - 35 USC § 103

11. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are

> such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

12. Claims 1-23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Trimberger, U.S. Patent No. 5,737,631, in view of Haynes et al., “Configurable Multiplier Blocks for use within an FPGA”, 1998 (herein referred to as Haynes), and further in view of Lodi et al., "A Flexible LUT-Based Carry Chain for FPGAs", 2003 (herein referred to as Lodi).

13. Referring to claim 1, Trimberger has taught a computer processor having control and data processing capabilities comprising:

a) a decode unit for decoding instructions (Trimberger: Figure 2, item 112).

b) a dedicated data processing facility having its own data register file (Fig.2, component 103 or 130), the data processing facility comprising a first data execution path including fixed operators (Trimberger: Figure 2, item 100) and a second data execution path including at least configurable operators (Trimberger: Figure 2, item 120) and a controller (controllers inherently exist in processors).

c) wherein said decode unit is operable to detect whether a data processing instruction defines a fixed data processing instruction or a configurable data processing instruction, said decode unit causing the computer processor to supply said data processing instruction to said first data execution path for processing when a fixed data processing instruction is detected and to said configurable data execution path for processing when a configurable data processing instruction is detected (Trimberger: column 7, lines 45-50).

d) Trimberger has not taught a dedicated control processing facility comprising a control execution path having its own control register file. However, Official Notice is taken that branch units pushing return addresses on a stack-based register file is a well known and advantageous

concept in the art. Specifically, a branch unit is a unit which controls program flow in response to a branch, such as a subroutine call or return instruction. Having these instructions is useful because it allows the programmer to repeat a code routine by simply calling the routine. Without a call/return, each time the routine is to be repeated, the actual routine would have to be duplicated. Consequently, by having call/return instructions, code density is increased by only having to write the routine once. Furthermore, in response to such a call, the branch unit will push a return address into its register file stack, which is also a well known component. In response to a return from subroutine instruction, an address would be retrieved from the top of the register file stack. Note that such a register file stack is known to be used in return address prediction which is clearly useful because a return address may be predicted by the register file stack before the true address is fetched from main memory, thereby allowing the processor to continue execution without stalling (which in turn increases throughput). As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Trimberger to include a branch unit having its own register file for executing call and return-type instructions, and predicting return addresses to increase throughput. It should further be noted that such a register file is only written to and read from when a branch of some sort occurs, and consequently, it can be said that the branch unit has its own register file.

e) Trimberger has also not taught said configurable operators pre-configured into a plurality of hardwired operator classes. However, both Haynes and Lodi have taught that FPGAs may be designed to include dedicated circuitry pre-configured into a plurality of hardwired operator classes. That is, Haynes has taught dedicated multiplier blocks, which may be programmably interconnected to create larger multipliers. Similarly, Lodi has taught dedicated carry chains,

which may be connected to form adders, for instance. See page 133 and section 3.1. Essentially, instead of a fully reprogrammable FPGA, as taught by Trimberger, Haynes and Lodia have taught programmable devices, such as FPGAs, that would include configurable operators pre-configured into hardwired multiplier and carry chain classes. The examiner asserts that it is well known in the art for FPGAs, such as the Xilinx Virtex, to include dedicated multipliers and carry chains. Such a configuration, with a programmable portion, and a dedicated portion, allows a programmer to achieve balance between programmability and speed for different applications. One of ordinary skill in the art would have recognized that the hardwired operators of Haynes and Lodi could be implemented in the FPGA of Trimberger. Such a modification would allow Trimberger to achieve both programmability and speed in cases where dedicated circuitry is faster than programmable circuitry. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Trimberger's gate array to include configurable operators pre-configured into a plurality of hardwired operator classes, as taught by Haynes and Lodi.

f) Trimberger, as modified, has further taught that said controller is operable to configure the connectivity of said configurable operators in accordance with configuration information provided in an opcode portion of said configurable data processing instruction. See column 3, lines 31-33, and note that Trimberger would still select the connectivity of the operator classes via program instruction opcode.

14. Referring to claim 2, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, wherein the decode unit is capable of decoding a stream of

instruction packets from memory, each packet comprising a plurality of instructions (Trimberger: column 7, lines 51-56).

15. Referring to claim 3, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, wherein the decode unit is operable to detect if an instruction packet contains a data processing instruction (Trimberger: column 7, lines 45-50).

16. Referring to claim 4, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, wherein the configurable operators are configurable at the level of multi-bit values (Trimberger: column 9, lines 18-19) (The opcode is a multi-bit value).

17. Referring to claim 5, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 4, wherein the configurable operators are configurable at the level of multi-bit values comprising four or more bits (Trimberger: column 9, lines 18-19) (The opcode is at least 4 bits).

18. Referring to claim 6, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 4, wherein the configurable operators are configurable at the level of words (Trimberger: column 9, lines 24-25) (Immediate values are optional; therefore, the whole word is configurable).

19. Referring to claim 7, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1. Trimberger has not taught that a plurality of the fixed operators of the first data execution path is arranged to perform a plurality of fixed operations in independent lanes according to single instruction multiple data principles. However, Official Notice is taken that SIMD and the related advantages are well known and accepted in the art. Specifically, SIMD allows each execution unit to perform the same instruction on different data

in the same cycle, thereby increasing data level parallelism. Higher parallelism will potentially result in higher throughput as more operations can occur at once. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Trimberger such that a plurality of the fixed operators of the first data execution path (Fig.2, component 100) is arranged to perform a plurality of fixed operations in independent lanes according to single instruction multiple data principles.

20. Referring to claim 8, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1. Trimberger in view of Haynes and Lodi has not taught that a plurality of the configurable operators of the second data execution path is arranged to perform multiple operations in different lanes according to single instruction multiple data principles. However, Official Notice is taken that SIMD and the related advantages are well known and accepted in the art. Specifically, SIMD allows each execution unit to perform the same instruction on different data in the same cycle, thereby increasing data level parallelism. Higher parallelism will potentially result in higher throughput as more operations can occur at once. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Trimberger in view of Haynes and Lodi such that a plurality of the configurable operators of the second data execution path (Trimberger, Fig.2, component 120) is arranged to perform multiple operations in different lanes according to single instruction multiple data principles.

21. Referring to claim 9, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, wherein configurable operators of the second execution path are

arranged to receive configuration information which determines the nature of the operations performed (Trimberger: column 8, lines 5-17).

22. Referring to claim 10, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 9, wherein configurable operators of the second execution path are arranged to receive configuration information which determines the nature of the operations performed from a field of an instruction defining a configurable data processing operation (Trimberger: column 7, lines 62-67; column 8, lines 1-17).

23. Referring to claim 11, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, wherein configurable operators of the second execution path are arranged to receive configuration information comprising information controlling connectivity of the configurable operators (Trimberger: column 8, lines 35-37).

24. Referring to claim 12, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 9, comprising a control map associated with configurable operators of the second data execution path, said control map being operable to receive at least one configuration bit from a configurable data processing instruction and to provide configuration information to the configurable operators responsive thereto (Trimberger: column 7, lines 62-67; column 8, lines 1-17).

25. Referring to claim 13, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 12, wherein said configuration information controls interconnectivity between two or more of said configurable operators (Trimberger: column 8, lines 35-37).

26. Referring to claim 14, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, wherein configurable operators of the second execution path are arranged to receive either configuration information determining the nature of an operation to be performed or configuration information controlling interconnectivity from a source other than a configurable data processing instruction (Trimberger: column 8, lines 5-17; Figure 2; items 101, 102 and 123).

27. Referring to claim 15, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, wherein at least one configurable operator of the second data execution path is capable of executing data processing instructions with an execution depth greater than two computations before returning results to a results store (Trimberger: column 3, lines 10-27) (It is inherent that these complex functions will take at least two cycles).

28. Referring to claim 16, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, comprising a switch mechanism for receiving data processing operands from a configurable data processing instruction and switching them as appropriate for supply to one or more of said configurable operators (Trimberger: column 7, lines 45-50).

29. Referring to claim 17, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, comprising a switch mechanism for receiving results from one or more of said configurable operators and switching the results as appropriate for supply to one or more of a result store and feed back loop (Trimberger: column 8, lines 51-59).

30. Referring to claim 18, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, comprising a plurality of control maps for mapping configuration bits received from configurable data processing instructions to configuration information for

supply to configurable operators of the second data execution path (Trimberger: column 3, lines 66-67; column 4, lines 1-10).

31. Referring to claim 19, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, comprising a switch mechanism for receiving configuration information from a control map and switching it as appropriate for supply to configurable operators of the second data execution path (Trimberger: column 8, lines 5-17).

32. Referring to claim 20, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, comprising configurable operators selected from one or more of: multiply accumulate operators; arithmetic operators; state operators; and cross-lane permuters (Trimberger: column 3, lines 10-27).

33. Referring to claim 21, Trimberger in view of Haynes and Lodi has taught a computer processor according to claim 1, comprising operators and an instruction set capable of performing one or more operations selected from: Fast Fourier Transforms; Inverse Fast Fourier Transforms; Viterbi encoding/decoding; Turbo encoding/decoding; and Finite Impulse Response calculations; and any other Correlations or Convolutions (Trimberger: column 3, lines 10-27) (polynomial evaluation is used in FFT and IFFT).

34. Referring to claim 22, Trimberger has taught a method of operating a computer processor having control and data processing capabilities, said computer processor comprising:

a) a decode unit for decoding instructions (Fig.2, component 112).

b) a dedicated data processing facility having its own data register file (Fig.2, component 103 or 130), the data processing facility comprising:

b1) a first data execution path including fixed operators (Fig.2, component 100).

b2) a second data execution path including at least configurable operators (Fig.2, component 120) and a controller (controllers are inherently present in processors), the method comprising:

- decoding a plurality of instructions to detect whether at least one data processing instruction, of said plurality of instructions, defines a fixed data processing instruction or a configurable data processing instruction (Trimberger: column 7, lines 45-50).
- causing the computer processor to supply said data processing instruction to said first data execution path for processing when a fixed data processing instruction is detected and to said configurable data execution path for processing when a configurable data processing instruction is detected; and outputting results produced by said execution path (Trimberger: column 7, lines 45-50).

c) Trimberger has not taught a dedicated control processing facility comprising a control execution path having its own control register file. However, Official Notice is taken that branch units pushing return addresses on a stack-based register file is a well known and advantageous concept in the art. Specifically, a branch unit is a unit which controls program flow in response to a branch, such as a subroutine call or return instruction. Having these instructions is useful because it allows the programmer to repeat a code routine by simply calling the routine. Without a call/return, each time the routine is to be repeated, the actual routine would have to be duplicated. Consequently, by having call/return instructions, code density is increased by only having to write the routine once. Furthermore, in response to such a call, the branch unit will

push a return address into its register file stack, which is also a well known component. In response to a return from subroutine instruction, an address would be retrieved from the top of the register file stack. Note that such a register file stack is known to be used in return address prediction which is clearly useful because a return address may be predicted by the register file stack before the true address is fetched from main memory, thereby allowing the processor to continue execution without stalling (which in turn increase throughput). As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Trimberger to include a branch unit having its own register file for executing call and return-type instructions, and predicting return addresses to increase throughput. It should further be noted that such a register file is only written to and read from when a branch of some sort occurs, and consequently, it can be said that the branch unit has its own register file.

d) Trimberger has also not taught said configurable operators pre-configured into a plurality of hardwired operator classes. However, both Haynes and Lodi have taught that FPGAs may be designed to include dedicated circuitry pre-configured into a plurality of hardwired operator classes. That is, Haynes has taught dedicated multiplier blocks, which may be programmably interconnected to create larger multipliers. Similarly, Lodi has taught dedicated carry chains, which may be connected to form adders, for instance. See page 133 and section 3.1. Essentially, instead of a fully reprogrammable FPGA, as taught by Trimberger, Haynes and Lodia have taught programmable devices, such as FPGAs, that would include configurable operators pre-configured into hardwired multiplier and carry chain classes. The examiner asserts that it is well known in the art for FPGAs, such as the Xilinx Virtex, to include dedicated multipliers and carry chains. Such a configuration, with a programmable portion, and a dedicated portion, allows a

programmer to achieve balance between programmability and speed for different applications. One of ordinary skill in the art would have recognized that the hardwired operators of Haynes and Lodi could be implemented in the FPGA of Trimberger. Such a modification would allow Trimberger to achieve both programmability and speed in cases where dedicated circuitry is faster than programmable circuitry. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Trimberger's gate array to include configurable operators pre-configured into a plurality of hardwired operator classes, as taught by Haynes and Lodi.

e) Trimberger, as modified, has further taught configuring the connectivity of said configurable operators in accordance with configuration information provided in an opcode portion of said configurable data processing instruction. See column 3, lines 31-33, and note that Trimberger would still select the connectivity of the operator classes via program instruction opcode.

35. Referring to claim 23, claim 23 is rejected for the same reasons set forth in the rejection of claim 22.

Response to Arguments

36. Applicant's arguments with respect to claims 1-23 have been considered but are moot in view of the new ground(s) of rejection.

37. Regarding the traversal of the examiner's official taking of Official Notice that a branch unit is well known to push addresses on a stack-based register file (on pages 13-14 of the remarks), the examiner would like to direct applicant's attention to Moshier, U.S. Patent No. 4,228,498, which is herein cited as extrinsic evidence for supporting the examiner's claim.

Specifically, see column 10, lines 8-23 and note that a register file is configured into a LIFO stack for holding multiple return address entries for nested subroutine support. Because the examiner has supplied a reference supporting the taking of Official Notice, the taking of Official Notice is maintained.

Conclusion

38. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection of claims, the patentable novelty must be clearly shown in view of the state of the art disclosed by the references cited and the objections made. Applicant must also show how the amendments avoid such references and objections. See 37 CFR § 1.111(c).

Douglass et al., U.S. Patent No. 6,798,239, has taught a programmable gate array having interconnecting logic to support embedded fixed logic circuitry.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to DAVID J. HUISMAN whose telephone number is (571)272-4168. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/David J. Huisman/
Primary Examiner, Art Unit 2183
January 12, 2009